OIPE JC99
APR 19 2004
PATENT & TRADEMARK OFFICE

MAIL STOP APPEAL BRIEF - PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants:	D.A. Sobeski et al.	Attorney Docket No. MSFT117220
Application No:	09/199,604	Group Art Unit: 2126
Filed:	November 25, 1998	Examiner: S. Courtenay III
Title:	DYNAMIC OBJECT BEHAVIOR FOR OBJECT-ORIENTED COMPUTING ENVIRONMENTS	

TRANSMITTAL OF APPEAL BRIEF
PETITION FOR EXTENSION OF TIME

RECEIVED
APR 21 2004
Technology Center 2100

Seattle, Washington 98101

April 16, 2004

TO THE COMMISSIONER FOR PATENTS:

A. Enclosed herewith for filing in the above-identified application is an Appeal Brief in triplicate. The enclosed check includes the requisite fee of $330.

B. Petition for Extension of Time

Applicants respectfully request that the period for filing an Appeal Brief, set to expire two months following the filing of a Notice of Appeal on December 19, 2003, be extended by 2 months, to expire on April 19, 2004. The enclosed check includes the one-month extension fee of $420.

C. Fees Enclosed

Enclosed is our Check No. 154998 in the amount of $750 to cover the amendment claim fee and request for extension of time fee.

The Commissioner is hereby authorized to charge any fees under 37 C.F.R. §§ 1.16, 1.17 and 1.18 which may be required during the entire pendency of the application, or credit any overpayment, to Deposit Account No. 03-1740. This authorization also hereby includes a request

04/20/2004 MAHMED1 00000043 09199604
02 FC:1252 420.00 OP

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS[PLLC]
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

for any extensions of time of the appropriate length required upon the filing of any reply during the entire prosecution of this application. A copy of this sheet is enclosed.

Respectfully submitted,

CHRISTENSEN O'CONNOR
JOHNSON KINDNESS^PLLC

Tracy S. Powell
Registration No. 53,479
Direct Dial No. 206.695.1786

I hereby certify that this correspondence is being deposited with the U.S. Postal Service in a sealed envelope as first class mail with postage thereon fully prepaid and addressed to Mail Stop Appeal Brief - Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the below date.

Date: April 16, 2004

TSP:lal

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^PLLC
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

MSFT\17220197.DOC

C E JC99
APR 19 2004
PATENT & TRADEMARK OFFICE

MAIL STOP APPEAL

BRIEF - PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Applicants:	D.A. Sobeski et al.	Attorney Docket No. MSFT117220
Application No:	09/199,604	Group Art Unit: 2126
Filed:	November 25, 1998	Examiner: S. Courtenay III
Title:	DYNAMIC OBJECT BEHAVIOR FOR OBJECT-ORIENTED COMPUTING ENVIRONMENTS	

RECEIVED
APR 21 2004
Technology Center 2100

APPELLANTS' APPEAL BRIEF

Seattle, Washington

April 16, 2004

TO THE COMMISSIONER FOR PATENTS:

This brief is in support of a Notice of Appeal filed in the above-identified application on December 19, 2003, to the Board of Patent Appeals and Interferences appealing the decision dated September 30, 2003, of the Primary Examiner finally rejecting Claims 1-28.

04/20/2004 MAHMED1 00000043 09199604
01 FC:1402 330.00 OP

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS PLLC
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

RECEIVED
APR 2 1 2004
Technology Center 2100

TABLE OF CONTENTS

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^PLLC
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

TABLE OF AUTHORITIES

Page

FEDERAL CASES

FEDERAL STATUTES

MISCELLANEOUS

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS[PLLC]
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

I. REAL PARTY IN INTEREST

The subject application is owned by Microsoft Corporation of Redmond, Washington.

II. RELATED APPEALS AND INTERFERENCES

Upon information and belief, appellants do not have any knowledge of related appeals or interferences that may directly effect or have a bearing on the decision of the Board of Appeals and Interferences (hereinafter "the Board") in the pending appeal.

III. STATUS OF THE CLAIMS

On November 25, 1998, appellants filed the pending patent application including Claims 1-19. On June 25, 2002, the Examiner issued a first Office Action rejecting Claims 1-19. On September 25, 2002, appellants filed an amendment and response in which no claims were amended, no claim was canceled, and no new claims were added.

On October 21, 2002, the Examiner issued a second Office Action finally rejecting Claims 1-19. On December 20, 2002, appellants filed an amendment and response in which Claims 1, 6, 9, and 14-18 were amended, Claim 20 was added, and no claims were canceled.

On January 7, 2003, the Examiner issued an advisory action refusing to enter the amendments and response on the basis that they raised new issues that would require further search and consideration. On January 21, 2003, appellants filed a Request for Continued Examination, with the January 21, 2003 response, resubmitting the response and un-entered amendments filed on December 20, 2002. Claims 1, 6, 9, and 14-18 were amended, Claim 20 was added, and no claims were canceled.

On March 6, 2003, the Examiner issued a first Office Action after the Request for Continued Examination rejecting Claims 1-20. On August 29, 2003, appellants filed an amendment and response in which Claims 1, 9, and 16-18 were amended, Claims 21-28 were newly added, and no claims were canceled.

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^PLLC
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

On September 30, 2003, the Examiner issued an Office Action finally rejecting Claims 1-28. On December 19, 2003, appellants filed a Notice of Appeal in which appellants request the Board to reverse the final rejections of Claims 1-28. The claims on appeal, Claims 1-28, are set forth in Section IX, Appendix of Claims Involved in the Appeal.

IV. STATUS OF AMENDMENTS

Appellants have not submitted any claim amendments subsequent to the September 30, 2003, Office Action's final rejections of pending Claims 1-28.

V. SUMMARY OF THE INVENTION

Prior to discussing appellants' invention, a brief background of the invention is provided to help the Board better appreciate appellants' invention discussed thereafter.

Background of the Present Invention: Objects in an Object-Oriented Environment

In an object-oriented (computing) environment, an "object" is a self-contained entity, comprising both data and routines, that is capable of interacting with other objects in the object-oriented environment. Objects in the object-oriented environment communicate with each other using messages, which, in essence, are calls to an object's exposed routines.

Most objects do not simply exist within an object-oriented environment. Instead, an instance of an object must be created. Creating an instance of an object is called instantiation, or instantiating an object. Only when an object is instantiated can it respond according to its defined behaviors with other objects in the object-oriented environment.

According to current practices, the behaviors of a single object, including the data stored by an object in its data structures, an object's response to messages sent from other objects or system events, and the actions performed when an object is instantiated, are all defined by a software programmer during a so-called coding process. In order to modify any of the defined behaviors of an object, the source code of that object must be modified to reflect the changes, the

code must then be recompiled, and the modified object must ultimately be re-deployed in an object-oriented environment. Because behavioral changes must be made in code, and once programmed in code they become fixed, the behaviors may be referred to as "static" behaviors.

As an example, assume that Object O has been programmed and compiled to perform Behaviors A and B. In order to add Behavior C to Object O, the source code for Object O must be reprogrammed to include Behavior C. After recompiling the source code for Object O, now coded to perform Behaviors A, B, and C, Object O can be redeployed in the object-oriented environment. Similarly, once Object O is deployed with defined Behaviors A, B, and C, Object O's source code must be modified, recompiled, and redeployed to eliminate Behavior C.

With the goal of having objects that exhibit dynamic behaviors, one prior art solution has been to simulate dynamic behavior using a technique frequently referred to as clustering. At the heart of this technique is a cluster object. One of the cluster object's statically defined behaviors is to create a structure, called a cluster, in which other objects, sometimes referred to as role objects, may be dynamically added and/or removed. Just as with the cluster object, role objects have specific, statically defined behaviors. One of these static behaviors is to interact with a cluster object. The cluster object's defined behavior also represents the role objects added to the cluster as a single entity. Thus, the behaviors of the cluster object, as well as each role object in the cluster, are fixed, statically defined behaviors. Dynamically adding and/or removing role objects to the cluster simulates dynamic behavioral changes, i.e., the ability to change the cluster's behaviors without requiring reprogramming and recompilation.

Summary of the Present Invention

Appellants' invention is directed at providing dynamic object behavior in individual objects. Each object, formed in accordance with the present invention includes a plurality of defined dynamic behaviors. (Specification, page 1, lines 25-26.) Information regarding the object's defined behaviors is stored in a data store. (Specification, page 1, line 26 - page 2,

line 1.) An application obtains the information regarding the defined behaviors from the data store and instantiates an object according to this information. (Specification, page 2, lines 1-2.) In contrast to the prior art described in the foregoing Background, defined dynamic behaviors may be added and/or removed by modifying the information in the data store, without requiring recoding and recompilation of the object's source code. (Specification, page 6, lines 20-22.) Thus, in operation, after instantiating a first object, the object's defined behaviors may be modified in the data store such that a second instance of the object will exhibit different behaviors than the first.

According to additional aspects of the present invention, an object may also determine, at run-time, which of its defined behaviors to exhibit/implement. (Specification, page 6, lines 18-20.) Depending on various conditions, such as specific system environment conditions, an instantiated object may implement less than all of its defined dynamic behaviors in the data store. (Specification, page 6, lines 27-29.) Thus, in operation, after instantiating a first instance of an object, a system environmental condition may be set such that a second instance of the object will exhibit different dynamic behaviors than the first.

As will be appreciated from the above description, the present invention provides dynamism in object behaviors in at least two manners. (Specification, page 6, lines 20-22.) First, the defined behaviors of the object can be dynamically modified in the data store, and second, the object is able to determine which of the dynamic behaviors defined in the data store to instantiate/exhibit at run-time, both without requiring recoding and recompiling. This represents a substantial improvement over the static nature of objects in the prior art, where behaviors of individual objects are altered only through a recoding and recompilation process.

Explanation of the Invention as Defined in the Claims

Independent Claim 1 is directed to a system. (Specification, pages 6-8, and FIGURES 1-2.) The system comprises a **single** object having a plurality of dynamic behaviors, a

data store to store data regarding the plurality of dynamic behaviors, and an application to instantiate the (single) object from the data stored in the data store regarding the plurality of dynamic behaviors. (Specification, page 6, lines 14-17, and FIGURE 2.) Further, when instantiated, the (single) object instantiates at least one of the plurality of dynamic behaviors. (Specification, page 6, lines 18-22.)

Claims 2-8 are dependent from independent Claim 1 and are directed to further limitations of the system described above. Claim 2 is dependent on Claim 1 and recites that the data store is within the (single) object. (Specification, page 6, lines 15-17, and page 10, original Claim 2.) Claim 3 is dependent from Claim 2 and recites that less than all of the plurality of dynamic behaviors of the (single) object are instantiated. (Specification, page 6, lines 27-29.) Claim 4 is dependent from Claim 2 and recites that the (single) object determines, at run time, which of the plurality of dynamic behaviors to instantiate. (Specification, page 6, lines 20-22.) Claim 5 is dependent from independent Claim 1 and recites that the (single) object comprises a Component Object Model (COM) object. (Specification, page 7, lines 10-13.)

Claim 6 depends from Claim 1 and recites that the plurality of dynamic behaviors comprises at least one selected from the group essentially consisting of a plurality of methods and a plurality of events. (Specification, page 7, lines 10-11.) Claim 7 is dependent from independent Claim 1 and recites the data store comprises the registry. (Specification, page 6, lines 14-17.) Claim 8 depends from Claim 1 and recites the plurality of dynamic behaviors comprises a plurality of system-defined behaviors and a plurality of application-defined behaviors. (Specification, page 7, lines 23-30.) Claim 20 depends from independent Claim 1 and further limits Claim 1 by reciting that the (single) object instantiates at least one of the plurality of dynamic behaviors during instantiation of the object. (Specification, page 6, lines 25-27.)

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS[PLLC]
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

Independent Claim 9 is directed to a method. (Specification, pages 8-9, and FIGURE 3.) According to the recited method, a command to instantiate a first instance of a **single** object having a plurality of dynamic behaviors associated with the (single) object is received. (Specification, page 8, lines 18-19, FIGURE 3, box 300.) Data regarding the plurality of dynamic behaviors in a data store is looked up. (Specification, page 8, lines 20-21; FIGURE 3, box 302.) Thereafter, the first instance of the (single) object from the data regarding the plurality of dynamic behaviors in the data store is instantiated. (Specification, page 8, lines 23-24; FIGURE 3, box 304.)

Claims 10-15 depend from independent Claim 9 and are directed to further limitations of the method described above. Claim 10 is dependent from Claim 9 and recites changing the plurality of dynamic behaviors. (Specification, page 8, line 26; FIGURE 3, box 306.) Claims 11 and 12 recite changing the plurality of dynamic behaviors by deleting (Claim 11) or adding (Claim 12) a dynamic behavior to the plurality of dynamic behaviors. (Specification, page 8, lines 25-27.) Claim 13 is dependent from Claim 10 and further recites changing the dynamic behaviors by changing the data stored in the data store regarding the plurality of dynamic behaviors. (Specification, page 8, lines 27-29.) Claim 14 is dependent from Claim 10 and further recites looking up data regarding the plurality of dynamic behaviors in the data store as have been changed, and instantiating a second instance of the (single) object from the data regarding the plurality of dynamic behaviors as have been stored in the data store, wherein the first and second instances of the (single) object exhibit different dynamic behaviors. (Specification, page 9, lines 1-6; FIGURE 3, boxes 308 and 310.) Claim 15 is dependent from Claim 14 and further comprises instantiating a data providing (single) object to provide data regarding the plurality of dynamic behaviors, and instantiating at least one instance of the (single) object from the data regarding the plurality of dynamic behaviors. (Specification, page 11; original Claim 15.)

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS[PLLC]
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

Independent Claim 16 is directed to a computer-readable medium having data stored thereon. (Specification, original Claim 16.) The data includes a **single** object having a plurality of dynamic behaviors associated with the (single) object, a data store to store data regarding the plurality of dynamic behaviors, and an application to instantiate the (single) object from the data stored in the data store regarding the plurality of dynamic behaviors. (Specification, page 8, lines 1-4.)

Independent Claim 17 is directed to a computer-readable medium having a computer program stored thereon for execution on a computer. The program stored on the computer-readable medium performs the method comprising receiving a command to instantiate a first instance of a **single** object having a plurality of dynamic behaviors associated with the (single) object, looking up data regarding the plurality of dynamic behaviors in a data store, instantiating the first instance of the (single) object from the data regarding the plurality of dynamic behaviors in the data store, changing the plurality of dynamic behaviors, looking up data regarding the plurality of dynamic behaviors in the data store as having been changed, and instantiating a second instance of the (single) object from the data regarding the plurality of dynamic behaviors as have been changed stored in the data store, wherein the first and second instances of the (single) object do not exhibit the same dynamic behaviors. (Specification, pages 8-9, page 12, and original Claim 17.)

Claim 18 is directed to a computer. (Specification, FIGURE 1.) The computer comprises a memory, a processor, a data store of the memory to store data regarding a plurality of dynamic behaviors defined for a **single** object, and an application executed by the processor from the memory to instantiate the (single) object from the data stored in the data store regarding the plurality of dynamic behaviors. (Specification, pages 6-8, page 12; original Claim 18; and FIGURES 1 and 2.)

Claim 19 depends from Claim 18 and further limits Claim 18 by reciting that the (single) object comprises a Component Object Model (COM) object, and the data store comprises the registry. (Specification, page 7, lines 9-15.)

Independent Claim 21 is directed to a system. The system includes a **single** object having an interface, and also having a plurality of dynamic behaviors, wherein the interface and the plurality of dynamic behaviors are defined for the (single) object (Specification, page 7, lines 1-16); a data store to store data regarding the plurality of dynamic behaviors (Specification, page 7, lines 9-16); and an application to instantiate the (single) object from the data stored in the data store regarding the plurality of dynamic behaviors (Specification, page 8, lines 8-16), and wherein the (single) object instantiates at least one of the plurality of dynamic behaviors. (Specification, page 9, lines 7-14.)

Claims 22-28 depend from independent Claim 21, and are directed to further limitations of the system described above. Claim 22 is dependent from Claim 21 and recites that the data store is within the (single) object. (Specification, page 6, lines 15-17, and page 10, original Claim 2.) Claim 23 is dependent from Claim 22 and recites that less than all of the plurality of dynamic behaviors of the (single) object are instantiated. (Specification, page 6, lines 27-29.) Claim 24 is dependent from Claim 22 and recites that the (single) object determines, at run time, which of the plurality of dynamic behaviors to instantiate. (Specification, page 6, lines 20-22.) Claim 25 is dependent from independent Claim 21, and recites that the (single) object comprises a Component Object Model (COM) object. (Specification, page 7, lines 10-13.)

Claim 26 depends from independent Claim 21 and recites that the plurality of dynamic behaviors comprises at least one selected from the group essentially consisting of a plurality of methods and a plurality of events. (Specification, page 7, lines 10-11.)

Claim 27 is dependent from independent Claim 21 and recites the data store comprises the registry. (Specification, page 6, lines 14-17.)

Claim 28 also depends from independent Claim 21 and recites that the plurality of dynamic behaviors comprises a plurality of system defined behaviors and a plurality of application defined behaviors. (Specification, page 7, lines 23-30.)

VI. ISSUES PRESENTED FOR REVIEW

In the September 30, 2003, final Office Action (hereinafter "Office Action") in this patent application, the Examiner rejected Claims 1-28 under 35 U.S.C. § 103 as being unpatentable over U.S. Patent No. 6,088,739, to Pugh et al. (hereinafter "Pugh") in view of *The Dynamics of Modeling Change*, to Corkill, Daniel D. (hereinafter "Corkill"). Additionally, Claims 1-28 were rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over Claims 1-19 of copending application No. 09/200,674, now U.S. Patent No. 6,304,879 to Sobeski et al. (hereinafter "Sobeski").

As will be discussed below, the Office Action fails to properly establish a *prima facie* case of obviousness in rejecting the pending claims. To establish a *prima facie* case of obviousness in rejecting a claim, each element of the claim must be taught or suggested by the prior art, M.P.E.P. § 2143.03. As described more particularly below, the cited and applied references fail to teach or suggest a **single** object having a plurality of dynamic behaviors, the plurality of dynamic behaviors being defined behaviors of the (single) object. In order to provide a basis for understanding and appreciating the arguments set forth below, appellants present a summary of the cited and applied 35 U.S.C. § 103 references.

Summary of Pugh (U.S. Patent No. 6,088,739)

Pugh describes an object clustering system for dynamically modifying the behavior of a simulated entity. (Pugh, abstract.) At the heart of this simulated entity is a cluster object that interfaces with and coordinates other objects referred to as role objects. The cluster object includes the ability to add and/or remove role objects to the simulated entity. For their part, each

role object is associated with a fixed, specific behavior. The cluster object exposes the fixed behaviors of each role object included in the collective, simulated entity. In other words, the cluster object exhibits static behaviors including adding and removing role objects, and each role object also exhibits a static role-type behavior. **Singly**, these objects do not exhibit dynamic behaviors. However, by adding and removing role objects to the cluster, the collective entity, i.e., the collection of **multiple** objects, exhibits dynamic behaviors. (Pugh, Col. 5, line 50 - Col. 6, lines 19).

Clearly, according to Pugh, while the behaviors of the collection of objects may be modified by adding and removing objects to the cluster, each individual object has only fixed, static behaviors. Pugh does not address dynamically changing the behavior of a **single** object, such as one of the role objects or the cluster object.

Summary of Corkill (*The Dynamics of Modeling Change*)

Corkill describes a so-called "blackboard" system to which various objects may be added and removed to develop a solution to a particular problem. According to Corkill, this "blackboard" paradigm permits people to engage in a brainstorming style interaction such that by adding and removing objects to a blackboard, a solution for a particular problem is dynamically obtained. (Corkill, p. 43, second column.)

Corkill makes liberal use of the adjective "dynamic," both in describing the blackboard system and the objects that are used by the blackboard system, and reflects an industry-wide abuse/overuse of the adjective when referring to object-oriented objects and programming in general. However, in regard to Corkill's so-called "dynamic" objects, Corkill discloses that these individual objects, with their static behaviors, may be dynamically linked into the blackboard system, including blackboard applications and knowledge stores. Thus, Corkill is, in many ways, similar to the Pugh system above, namely that individual objects, each with fixed behaviors, may be dynamically added to and/or removed from a collection of objects to simulate dynamic

behavior for the collection of objects. However, Corkill does not disclose, teach, or suggest that individual objects in and of themselves exhibit dynamic behaviors.

VII. GROUPING OF CLAIMS

Claims 1-8, and 20 stand or fall together; Claims 9-15, and 19 stand or fall together; Claim 16 stands or falls alone; Claim 17 stands or falls alone; Claim 18 stands or falls alone; and Claims 21-28 stand or fall together. The reasons why the grouped claims are believed to be separately patentable are explained below in the Argument.

VIII. ARGUMENTS

35 U.S.C. § 103 Rejections

As will be discussed in greater detail below, the claims of the present application are clearly and patentably distinguishable over the teachings of the above-cited references. The present invention is directed toward providing objects that exhibit dynamic behaviors. This dynamism is achieved in two ways: (1) the behaviors of an instance of a **single** object are determined at instantiation of the (single) object according to information stored in a behavior store, and (2) an individual (single) object may determine, during the instantiation of the object, which behaviors to exhibit according to system settings. Providing an object with dynamic behaviors represents a substantial improvement over the prior art, where individual prior-art objects have statically defined behaviors, and where dynamic behaviors are simulated using a collection of objects.

As noted above, the Office Action rejected Claims 1-28 under 35 U.S.C. § 103 as being unpatentable in view of the teachings of Pugh and Corkill. Appellants respectfully disagree. As discussed in more detail below, appellants assert that the cited and applied references fail to teach each element of the independent claims, much less each element of the dependent claims. These

claims, which particularly point out and distinctly claim subject matter that appellants regard as their invention, are clearly and patentably distinguishable from the cited and applied references.

Claims 1-8 and 20

Pugh and Corkill Fail to Teach or Suggest a Single Object With a Plurality of Dynamic Behaviors

The Office Action asserts that Pugh and Corkill teach or suggest each element of independent Claim 1. Appellants disagree, and submit that Pugh and Corkill, alone and in combination, fail to teach or suggest each element of independent Claim 1. In particular, Pugh and Corkill fail to teach or suggest "**a single object having a plurality of dynamic behaviors**, wherein the plurality of dynamic behaviors are defined behaviors of the object," as recited in Claim 1.

As previously discussed, Pugh purportedly discloses that a simulated entity has dynamic behaviors through an object clustering system. The simulated entity's dynamic behaviors are made possible by adding and removing role objects, each with fixed behaviors, to a clustering object, also having fixed behaviors. Clearly, Pugh discloses a system that requires the cooperation of at **least two objects** to simulate an entity with dynamic behaviors.

The Office Action asserts that Pugh discloses a single object having a plurality of dynamic behaviors. For support of this assertion, the Office Action quotes from Pugh that ". . . a composite object **appears** to a client of the composite object as a single object . . ." (Office Action, p. 2, citing Pugh, Col. 6, lines 65-66. Emphasis added.) That the simulated entity **appears** as a single object simply misses the point, namely that the simulated entity is a collection of multiple objects, and that the simulated entity exhibits dynamic behaviors by adding and removing statically defined role objects. In other words, while a simulated entity may appear externally as a single object with dynamic behaviors, the simulated entity is not a single object

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS[PLLC]
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

having dynamic behaviors. Clearly, the cited passage from Pugh represents an admission by Pugh that the cluster is not a single object.

The Office Action further asserts that Corkill "discloses the use of dynamic objects that are not composite objects," see Office Action, p. 2, and cites from Corkill that "[d]ynamic, multiple inheritance allows developers to augment generic object classes with specialized application information." (Corkill, p. 44, Col. 2.) As appellants previously mentioned in this brief, and as appellants also mentioned in at least one previous Office Action response, Corkill makes liberal use of the adjective "dynamic." However, Corkill never uses the adjective "dynamic" to refer to a single object having dynamic behaviors. In fact, the cited passage clearly indicates otherwise, namely that developers can "augment generic object classes with specialized application information." To those skilled in the art, this is a crystal clear reference to the code development process, particularly a developer coding an object class file to behave in a specific, augmented manner and compiling the edited code, the result being an object with augmented, but static, behaviors. Appellants assert that would be entirely unreasonable to those skilled in the art to interpret Corkill's statement as dynamic object behaviors as claimed in the present application.

Pugh and Corkill Fail to Teach or Suggest That the Plurality of Dynamic Behaviors Are Defined Behaviors of the Single Object.

The Office Action further rejected the recitation of Claim 1 that "the plurality of dynamic behaviors are defined behaviors of the object," stating that the "dynamic behaviors necessarily flow from the teaching of the art." (Office Action, p. 3.) Appellants assert that the Office Action misunderstands the recitation. In both Pugh and Corkill, to the extent that their systems disclose a collective/simulated entity having dynamic behaviors, the dynamic behaviors are not behaviors of the collective entity, but defined, static behaviors of the individual objects that are added to and/or removed from the collective entities. As such, neither Pugh nor Corkill disclose that the

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS[PLLC]
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

plurality of dynamic behaviors are defined behaviors of collective entity. Rather, they are behaviors of the objects that are added and/or removed from the collective entities.

A Proper *Prima Facie* Case of Obviousness Was Not Made

In order to establish a proper *prima facie* case of obviousness, three criteria must be met: (1) there must be some suggestion or motivation in the references or in the art generally to modify or combine the teachings, (2) there must be a reasonable expectation of success, and (3) **the prior art references must teach or suggest all the claim limitations**. (*In re Vaeck*, 947 F.2d 488, 20 U.S.P.Q.2d 1438 (Fed. Cir. 1991); M.P.E.P § 2143. Emphasis added). As described above, Pugh and Corkill, alone and in combination, clearly fail to teach or suggest each element of Claim 1, in particular, "a single object having a plurality of dynamic behaviors," and "wherein the plurality of dynamic behaviors are defined behaviors of the object." According to examination procedures, when an Office Action fails to establish a proper *prima facie* case of obviousness, applicants/appellants are under no obligation or burden to prove nonobviousness. (M.P.E.P. § 2142.) Accordingly, appellants assert that the 35 U.S.C. § 103(a) rejection of Claim 1 was in error. Appellants further assert that Claim 1 is allowable over Pugh and Corkill and request the reversal of the rejection of Claim 1 based on these references.

Claims 2-8 and 20 depend directly or indirectly from Claim 1. Accordingly, for the same reasons described above in regard to Claim 1, appellants assert that Pugh and Corkill, alone and in combination, fail to teach or suggest each element of dependent Claims 2-8 and 20, especially when read in conjunction with Claim 1. Appellants, therefore, assert that the 35 U.S.C. § 103(a) rejections of Claims 2-8 and 20 were in error, and that Claims 2-8 and 20 are allowable over Pugh and Corkill. Thus, reversal of the rejections of these claims based on Pugh and Corkill is also requested.

Claims 9-15, and 19

While Claim 1 is directed to a system, independent Claim 9 is directed to a method. Though directed at different subject matter, independent Claim 9 recites similar elements to those found in independent Claim 1. In light of these similarities, the Office Action rejected Claim 9 for similar reasons as to those expressed in rejecting Claim 1. Appellants disagree and assert that Pugh and Corkill, alone and in combination, fail to teach or suggest each element of Claim 9. In particular, Pugh and Corkill, alone and in combination, fail to teach or suggest "receiving a command to **instantiate a first instance of a single object having a plurality of dynamic behaviors,**" and "wherein the plurality of dynamic behaviors are defined behaviors of the object," as recited in Claim 9. (Emphasis added.)

For the same reasons as discussed above in regard to Claim 1, appellants assert that Pugh and Corkill, alone and in combination, fail to teach or suggest "**single object** having a plurality of dynamic behaviors." (Emphasis added.) Pugh discloses a collection of objects, i.e., more than a single object, that operate together as a simulated entity, and it is the simulated entity that exhibits dynamic behaviors through adding and removing role objects. Corkill discloses a similar technology where objects are added to and removed from a blackboard, analogous to the cluster object in Pugh, to dynamic create a solution to a particular problem (appellants again note that Corkill fails to disclose, teach, or suggest that the final solution exhibits dynamic behaviors.)

While Pugh and Corkill may disclose an aggregation of objects that collectively exhibit dynamic behaviors, neither Pugh nor Corkill teach or suggest a **single** object having a defined plurality of dynamic behaviors. Instead, each object added to the Pugh or Corkill system has fixed behaviors, and only as an aggregation of objects do their systems exhibit any form of dynamism.

As Pugh and Corkill fail to teach or suggest each element of Claim 9, appellants assert that a proper *prima facie* case of obviousness was not met. Accordingly, appellants assert that

the 35 U.S.C. § 103(a) rejection of Claim 9 was in error, and that Claim 9 is allowable over Pugh and Corkill. Thus, reversal of the rejection of Claim 9 based on Pugh and Corkill is also requested.

Claims 10-15 and 19, depend directly or indirectly from Claim 9. Accordingly, for the same reasons described above in regard to Claim 9, appellants assert that Pugh and Corkill, alone and in combination, fail to teach or suggest each element of dependent Claims 10-15 and 19, especially when read in conjunction with Claim 9. Appellants, therefore assert that the 35 U.S.C. § 103(a) rejections of Claims 10-15 and 19, were in error, and further assert that Claims 10-15 and 19, are allowable over Pugh and Corkill. Thus, reversal of these rejections is also requested.

Claim 16

Independent Claim 16 is directed to a computer-readable medium having data stored thereon. Though directed at different subject matter than Claim 1, independent Claim 16 recites similar elements to those found in independent Claim 1. In light of these similarities, the Office Action rejected Claim 16 for similar reasons as to those expressed in rejecting Claim 1. Appellants assert that Pugh and Corkill, alone and in combination, fail to teach or suggest each element of Claim 16. In particular, appellants assert that Pugh and Corkill, alone and in combination, fail to teach or suggest "**a single object having a plurality of dynamic behaviors,**" and "wherein the plurality of dynamic behaviors are defined behaviors of the object," as recited in Claim 16. (Emphasis added.)

For the same reasons as discussed above in regard to Claim 1, appellants assert that Pugh and Corkill, alone and in combination, fail to teach or suggest "**a single object** having a plurality of dynamic behaviors." (Emphasis added.) Pugh discloses a collection of objects, i.e., more than a single object, that operate together as a simulated entity, and it is the simulated entity that exhibits dynamic behaviors through adding and removing role objects. Corkill discloses a similar technology where objects are added to and removed from a blackboard, analogous to the

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^PLLC
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

cluster object in Pugh, to dynamic create a solution to a particular problem (again appellants note that Corkill fails to disclose, teach, or suggest that the final solution exhibits dynamic behaviors.)

While Pugh and Corkill may disclose an aggregation of objects that collectively exhibit dynamic behaviors, neither Pugh nor Corkill teach or suggest a **single** object having a defined plurality of dynamic behaviors. Instead, each object added to the Pugh or Corkill system has fixed behaviors, and only as an aggregation of objects do their systems exhibit any form of dynamism.

As Pugh and Corkill fail to teach or suggest each element of Claim 16, appellants assert that a proper *prima facie* case of obviousness was not met. Accordingly, appellants assert that the 35 U.S.C. § 103(a) rejection of Claim 16 was in error, and that Claim 16 is allowable over Pugh and Corkill. Reversal of this rejection is also requested.

<u>Claim 17</u>

Independent Claim 17 is directed to a computer-readable medium having a computer program stored thereon for execution on a computer, wherein the program performs a method. Though directed at different subject matter than independent Claims 1, 9, and 16, independent Claim 17 recites similar elements to those found in the independent claims. In light of these similarities, the Office Action rejected Claim 17 for similar reasons as to those expressed in rejecting Claim 1. Appellants assert that Pugh and Corkill, alone and in combination, fail to teach or suggest each element of Claim 17. In particular, appellants assert that Pugh and Corkill, alone and in combination, fail to teach or suggest "receiving a command to instantiate **a first instance of the single object having a plurality of dynamic behaviors,**" and "wherein the plurality of dynamic behaviors are defined behaviors of the object," as recited in Claim 17. (Emphasis added.)

For the same reasons as discussed above in regard to Claim 1, appellants assert that Pugh and Corkill, alone and in combination, fail to teach or suggest receiving a command to instantiate

"**a single object** having a plurality of dynamic behaviors." (Emphasis added.) Pugh discloses a collection of objects, i.e., more than a single object, that operate together as a simulated entity, and it is the simulated entity that exhibits dynamic behaviors through adding and removing role objects. Pugh fails to teach or suggest a single object having a plurality of dynamic behaviors, and therefore must also fail to teach or suggest receiving a command to instantiate such a single object.

Corkill discloses a similar technology to Pugh, where objects are added to and removed from a blackboard, analogous to the cluster object in Pugh, to dynamic create a solution to a particular problem (appellants again note that Corkill fails to disclose, teach, or suggest that the final solution exhibits dynamic behaviors.) Corkill fails to teach or suggest a single object having a plurality of dynamic behaviors. Therefore, Corkill must also fail to teach or suggest receiving a command to instantiate such a single object.

While Pugh and Corkill may disclose an aggregation of objects that collectively exhibit dynamic behaviors, neither Pugh nor Corkill teach or suggest a **single** object having a defined plurality of dynamic behaviors. Instead, each object added to the Pugh or Corkill system has fixed behaviors, and only as an aggregation of objects do their systems exhibit any form of dynamism.

As Pugh and Corkill fail to teach or suggest each element of Claim 17, appellants assert that a proper *prima facie* case of obviousness was not met. Accordingly, appellants assert that the 35 U.S.C. § 103(a) rejection of Claim 17 was in error, and that Claim 17 is allowable over Pugh and Corkill. Reversal of the rejection is thus also requested.

<u>Claim 18</u>

Independent Claim 18 is directed to a specific computer system. Though directed at different subject matter than Claims 1, 9, 16, and 17, independent Claim 18 recites similar elements to those found in independent Claim 1. In light of these similarities, the Office Action

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS^PLLC
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

rejected Claim 18 for similar reasons as to those expressed in rejecting the independent claims described above. Appellants assert that Pugh and Corkill, alone and in combination, fail to teach or suggest each element of Claim 18. In particular, appellants assert that Pugh and Corkill, alone and in combination, fail to teach or suggest "a data store of the memory to store data regarding **a plurality of dynamic behaviors defined for a single object**," as recited in Claim 16. (Emphasis added.)

While Pugh and Corkill may disclose a data store storing information regarding dynamic behaviors, the dynamic nature of the behaviors is in the aggregation of multiple objects, not a single object. Pugh and Corkill, alone and in combination, fail to teach or suggest a **single object** having a plurality of dynamic behaviors.

As Pugh and Corkill fail to teach or suggest each element of Claim 18, appellants assert that a proper *prima facie* case of obviousness was not met. Accordingly, appellants assert that the 35 U.S.C. § 103(a) rejection of Claim 18 was in error, and that Claim 18 is allowable over Pugh and Corkill. Reversal of the rejection is also requested.

<u>Claims 21-28</u>

Claims 21 is directed to a system. While similar in scope to Claim 1, Claim 21 includes an additional element, namely that the **single** object has an interface as well as a plurality of dynamic behaviors. The Office Action rejected Claim 21 for the same reasons as described in Claim 1 and additionally citing that Pugh discloses that the cluster object provides the ability to add a role interface to the cluster, i.e., the simulated entity. However, appellants note that the Office Action's rejection of Claim 1, in particular the Office Action's rejection of "a plurality of dynamic behaviors" was also based on Pugh's recitation of interfaces. In particular, the Office Action cited Pugh's statement: "composite object appears to a client of the composite object as a single object that exposes **multiple interfaces.**" (Office Action, p. 1, citing Pugh, Col. 5, lines 65-67.)

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS[PLLC]
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

Claim 21 clearly recites both an interface and a plurality of dynamic behaviors. Even assuming for purposes of argument that Pugh discloses a single object with an interface, as discussed above the Office Action utterly fails to cite any reference disclosing a single object having a plurality of dynamic behaviors. Instead, the Office Action impermissibly equates an interface to both an object's interface **and** an object's dynamic behaviors. (Pugh clearly does not disclose this subject matter.)

For the additional reasons described above, as well as those presented in regard to Claim 1, appellants assert that Pugh and Corkill fail to teach or suggest each element of Claim 21, and thus, a proper *prima facie* case of obviousness was not met. Accordingly, appellants assert that the 35 U.S.C. § 103(a) rejection of Claim 21 was in error, and that Claim 21 is allowable over Pugh and Corkill. Thus, reversal of the rejection is also requested.

Claims 22-28 depend directly or indirectly from Claim 21. Accordingly, for the same reasons described above in regard to Claim 21, appellants assert that Pugh and Corkill, alone and in combination, fail to teach or suggest each element of dependent Claims 22-28, especially when read in conjunction with Claim 21. Appellants, therefore assert that the 35 U.S.C. § 103(a) rejections of Claims 22-28 were in error, and further assert that Claims 22-28 are allowable over Pugh and Corkill. Reversal of these rejections is also requested.

Obviousness-Type Double Patenting Rejection

The Office Action also rejected pending Claim 1-28 under the judicially created doctrine of obviousness-type double patenting, as being unpatentable over Claim 1-19 of co-pending application 09/200,674, now U.S. Patent No. 6,304,879 to Sobeski et al. (hereinafter "Sobeski.") More particularly, the Examiner stated that while the claims between the present invention and Sobeski "are not identical, they are not patentably distinct from each because of corresponding language that recites many of the same elements and function claims in the previously patented

invention," and that "claimed differences would be obvious to a programmer of ordinary skill." (Office Action, pg. 13.)

The Office Action equates storing temporary data (dynamic properties) to dynamic object behaviors. Appellants disagree. The recited elements are not similar. Appellants further assert that even if similar, which appellants categorically deny, such similarity would be not obvious to one of ordinary skill in the art. Appellants assert that one of ordinary skill in the art readily recognize the clear distinction between how data is internally stored by object versus dynamic object behaviors.

Appellants also point to inconsistencies in the Office Action's treatment of the claims. In rejecting Claims 1-28 under 35 U.S.C. § 103(a), the Office Action takes the position that a collective/simulated entity may exhibit dynamic behaviors (actions). Yet in regard to the double-patenting rejection, the Office Action fails to distinguish between temporary data storage and dynamic object behaviors.

In view of the above, appellants assert that the rejection of Claims 1-28 under the judicially created doctrine of double-patenting was in error. Accordingly, appellants also request a reversal of this rejection.

Conclusion

In view of the foregoing remarks, appellants submit that all of the claims in the present application are clearly patentably distinguishable over the teachings of Pugh and Corkill. Appellants further submit that all of the claims in the present application are also patentably distinct from the claims of Sobeski. Therefore, it is submitted that the Examiner's rejections of Claims 1-28 were erroneous, and reversal of the decision is respectfully requested.

IX. APPENDIX OF CLAIMS INVOLVED IN THE APPEAL

1. A system comprising:

a single object having a plurality of dynamic behaviors, wherein the plurality of dynamic behaviors are defined behaviors of the object;

a data store to store data regarding the plurality of dynamic behaviors; and,

an application to instantiate the object from the data stored in the data store regarding the plurality of dynamic behaviors,

wherein the object instantiates at least one of the plurality of dynamic behaviors.

2. The system of claim 1, wherein the data store is within the object.

3. The system of claim 2, wherein less than all of the plurality of dynamic behaviors of the object are instantiated.

4. The system of claim 2, wherein the object determines at run-time which of the plurality of dynamic behaviors to instantiate.

5. The system of claim 1, wherein the object comprises a Component Object Model (COM) object.

6. The system of claim 1, wherein the plurality of dynamic behaviors comprises at least one selected from the group essentially consisting of a plurality of methods, and a plurality of events.

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS[PLLC]
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

7. The system of claim 1, wherein the data store comprises the Registry.

8. The system of claim 1, wherein the plurality of dynamic behaviors comprises a plurality of system-defined behaviors and a plurality of application-defined behaviors.

9. A method comprising:

receiving a command to instantiate a first instance of a single object having a plurality of dynamic behaviors, wherein the plurality of dynamic behaviors are defined behaviors of the object;

looking up data regarding the plurality of dynamic behaviors in a data store; and,

instantiating the first instance of the object from the data regarding the plurality of dynamic behaviors in the data store.

10. The method of claim 9, further comprising changing the plurality of dynamic behaviors.

11. The method of claim 10, wherein changing the plurality of dynamic behaviors comprises deleting one of the plurality of dynamic behaviors.

12. The method of claim 10, wherein changing the plurality of dynamic behaviors comprises adding a new dynamic behavior to the plurality of dynamic behaviors.

13. The method of claim 10, wherein changing the plurality of dynamic behaviors comprises changing the data stored in the data store regarding the plurality of dynamic behaviors.

14. The method of claim 10, further comprising:

looking up data regarding the plurality of dynamic behaviors in the data store as have been changed; and,

instantiating a second instance of the object from the data regarding the plurality of dynamic behaviors as have been stored in the data store, wherein the first and second instances of the object exhibit different dynamic behaviors.

15. The method of claim 14, further comprising:

instantiating a data providing object to provide data regarding the plurality of dynamic behaviors; and,

instantiating at least one instance of the object from the data regarding the plurality of dynamic behaviors.

16. A computer-readable medium having data stored thereon representing:

a single object having a plurality of dynamic behaviors, wherein the plurality of dynamic behaviors are defined behaviors of the object;

a data store to store data regarding the plurality of dynamic behaviors; and,

an application to instantiate the object from the data stored in the data store regarding the plurality of dynamic behaviors.

17. A computer-readable medium having a computer program stored thereon for execution on a computer, the program performing the method comprising:

receiving a command to instantiate a first instance of the single object having a plurality of dynamic behaviors, wherein the plurality of dynamic behaviors are defined behaviors of the object;

looking up data regarding the plurality of dynamic behaviors in a data store;

instantiating the first instance of the object from the data regarding the plurality of dynamic behaviors in the data store;

changing the plurality of dynamic behaviors;

looking up data regarding the plurality of dynamic behaviors in the data store as having been changed; and,

instantiating a second instance of the object form the data regarding the plurality of dynamic behaviors as have been changed stored in the data store, wherein the first and second instances of the object do not exhibit the same dynamic behaviors.

18. A computer comprising:

a memory;

a processor;

a data store of the memory to store data regarding a plurality of dynamic behaviors defined for a single object; and,

an application executed by the processor from the memory to instantiate the object from the data stored in the data store regarding the plurality of dynamic behaviors.

19. The computer of claim 14, wherein the object comprises a Component Object Model (COM) object, and the data store comprises the Registry.

20. The system of Claim 1, wherein the object instantiates at least one of the plurality of dynamic behaviors during instantiation of the object.

21. A system comprising:

a single object having an interface and also having a plurality of dynamic behaviors, wherein the interface and the plurality of dynamic behaviors are defined for the object;

a data store to store data regarding the plurality of dynamic behaviors; and,

an application to instantiate the object from the data stored in the data store regarding the plurality of dynamic behaviors, and

wherein the object instantiates at least one of the plurality of dynamic behaviors.

22. The system of claim 21, wherein the data store is within the object.

23. The system of claim 22, wherein less than all of the plurality of dynamic behaviors of the object are instantiated.

24. The system of claim 22, wherein the object determines at run-time which of the plurality of dynamic behaviors to instantiate.

25. The system of claim 21, wherein the object comprises a Component Object Model (COM) object.

26. The system of claim 21, wherein the plurality of dynamic behaviors comprises at least one selected from the group essentially consisting of a plurality of methods, and a plurality of events.

27. The system of claim 21, wherein the data store comprises the Registry.

28. The system of claim 21, wherein the plurality of dynamic behaviors comprises a plurality of system-defined behaviors and a plurality of application-defined behaviors.

Respectfully submitted,

CHRISTENSEN O'CONNOR
JOHNSON KINDNESS PLLC

Tracy S. Powell
Registration No. 53,479
Direct Dial No. 206.695.1786

I hereby certify that this correspondence is being deposited in triplicate with the U.S. Postal Service in a sealed envelope as first class mail with postage thereon fully prepaid and addressed to Mail Stop Appeal Brief - Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the below date.

Date: April 16, 2004   Lori A. Lewis

TSP:lal

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS PLLC
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

MSFT\17220199.DOC